

panel 286. Moreover, any changes made to a value, in say, a host group level 292 changes the inherited value of that property on its associated hosts (host3).

For example, selection of a particular policy 292 in the policy hierarchy structure 290 displayed in the first panel 286, results in the display of editable fields 300 in the second panel 288 that correspond to attributes 302, 304, 306 and inherited attribute values of that policy 292. Changes made by an operator/administrator to the threshold value 302, alert interval 304, and maximum file system size 306 in the second panel 288 are immediately reflected in the corresponding values 294, 296, 298 in the policy hierarchy structure 290 displayed in the first panel 286.

Moreover, the modifications made to items 294, 296, 298 are inherited by the associated hosts (host3) of that host group 292. In this instance, host3 inherits the alert interval 294, max file system size 296, and threshold 298 from host group 292. All the other attributes of host3 are inherited from the default values at the domain level of the policy hierarchy 290.

LUN Masking on Windows NT Hosts

As discussed above, storage devices are assigned to the host devices 12 by the manager 20, which effects those assignments using the agents on the respective host devices. Referring back to FIGURE 10 and the accompanying text, assigned LUN IDs are communicated to the hosts via the disk manager 76, which updates the filter drivers 79 on the respective hosts. When a host file system makes an attempt to mount a storage device, the filter driver 79 (FIGURE 10) intervenes, comparing an identifier of the device being mounted against the assigned LUN IDs. The driver

79 fails devices for which there is not a match and succeeds (or at least passes for normal treatment by the operating system) those for which there is a match.

FIGURE 36 depicts a storage driver architecture of the Windows™ NT operating system of an exemplary host 12 modified in accordance with the invention to provide these features, referred to elsewhere herein as "LUN masking."

The illustrated portion of the modified operating system 350 comprises a storage class driver 352 and port driver 356 of the conventional variety known and used in the art for the Windows™ NT operating system. In alternate embodiments, commercial or proprietary drivers providing like functionality can be used in addition or instead.

Generally, storage class driver 352 and port class driver 356 operate in the conventional manner to translate IRPs from the file system to appropriate form for transfer to the host bus adapter (see FIGURE 23, ADAPTER1 & ADAPTER2) associated with the attached storage devices (FIGURE 23, DISK1 - DISK3). More particularly, the storage class driver 352 uses the SCSI port/class interface to control one or more devices 14 on any bus for which the system provides a storage port driver 356. The port driver 356 serves as an interface between class drivers 352 and the host bus adapter (HBA) (FIGURE 23, ADAPTER1) that is connected to one or more storage devices (FIGURE 23, DISK1-DISK3). The SCSI port driver 356 receives SCSI request blocks (SRBs) from higher-level drivers (e.g., class driver, filter driver), and translates the SRBs into bus-specific commands that are then transferred to an HBA. An adapter-specific SCSI miniport driver is coupled to the port driver 356, and provides support for the particular SCSI HBA.

In the illustrated embodiment, filter driver 354 is interposed between storage class driver 352 and port driver 356. The filter driver includes a table 354a, or other data structure, listing LUN IDs that have been assigned to the associated host. The table is loaded and updated by the disk manager 76 (which typically communicates with the filter driver 354 and table 354a via a user mode applications program (not shown)) as discussed further below and can be persisted in a conventional manner, e.g., via a database or other persistent storage, not shown. (In alternate embodiments, rather than listing LUN IDs of devices that have been assigned to the host, the table 354a list LUN IDs of devices that from which access by the host to be blocked. Such alternate embodiments operate in the manner discussed herein, as appropriately modified to account for this difference in table content).

In normal operation, e.g., during boot-up of the Windows NT operating system or when the ports are otherwise scanned during system operation, the SCSI port driver 356 queries the SCSI bus to identify devices that are in communication with host 12. The port driver 356 then loads the SCSI addresses (each comprising multiple fields, e.g., port, bus, target id, logical unit number) of found devices (e.g., LUNs) into a port driver structure, and updates the Windows NT registry. The port driver also generates a physical device object for each identified device.

Continuing, in normal operation, the SCSI class driver 352 (a conventional Windows NT storage class driver) traverses the list of found device addresses, and issues claim requests to the port driver 356 for each of them. Normally, the SCSI port driver 356 responds to each of those requests by noting that the device is available to be claimed. The SCSI class driver 352 then